

METHOD AND APPARATUS FOR IN-SYSTEM PROGRAMMING
THROUGH A COMMON CONNECTION POINT OF PROGRAMMABLE
LOGIC DEVICES ON MULTIPLE CIRCUIT BOARDS OF A SYSTEM

RELATED APPLICATIONS

0341023-073001

[0001] This application is related to copending and cofiled applications for United States Letters Patent Serial No. _____, filed _____ and entitled SYSTEM AND METHOD FOR IN-SYSTEM PROGRAMMING THROUGH AN ON-SYSTEM JTAG BRIDGE OF PROGRAMMABLE LOGIC DEVICES ON MULTIPLE CIRCUIT BOARDS OF A SYSTEM (Attorney Docket No. 10016250-1); Serial No. _____, filed _____ and entitled METHOD FOR ACCESSING SCAN CHAINS AND UPDATING EEPROM-RESIDENT FPGA CODE THROUGH A SYSTEM MANAGEMENT PROCESSOR AND JTAG BUS (Attorney Docket No. 10017840-1); and Serial No. _____, filed _____ and entitled METHOD AND APPARATUS FOR SERIAL BUS TO JTAG BUS BRIDGE (Attorney Docket No. 10017841-1), all of the aforementioned applications incorporated herewith by reference thereto.

FIELD OF THE INVENTION

[0002] The invention is related to the art of providing configuration code information to programmable logic devices, including Field Programmable Gate Array (FPGA) devices in complex electronic system. In particular, the invention relates to methods and apparatus for programming FPGA code into EEPROM associated with FPGAs, or into EEPROM of FPGAS, through a central point in a system and over JTAG serial busses.

BACKGROUND OF THE INVENTION

[0003] Serial communications busses of the separate-clock-and-data type have become commonly used for communication between integrated circuit components of a system. Serial links of this type include the IIC (initially known as the Inter IC bus, now widely known as I2C) and SPI busses. Links of this type can be implemented without need of precision timing components at each integrated circuit on the bus and typically operate under control of at least one bus master. Serial EEPROM (Electrically Erasable

Programmable Read-Only Memory) devices are widely available that interface with serial communications busses of the SPI and IIC types.

[0004] While the I2C and SPI busses are typically used for communications within systems during normal operation, the IEEE 1149.1 serial bus, known as the JTAG bus, was intended for testing of inactive systems by providing access from a tester to perform a boundary scan on each integrated circuit. The tester can thereby verify connectivity of the integrated circuits and verify that they are installed and interconnected correctly. The JTAG bus provides for interconnection of one or more integrated circuits in a chain, any of which may be addressed by the tester. Typically, multiple devices of a circuit board are interconnected into a JTAG bus.

[0005] The JTAG bus uses four wires. These include a serial data-in line, a serial data-out line, a clock line, and a test mode select line. Typically the data-out line of a first chip in a chain couples to the data-in line of a second chip of the chain, and the data-out line of the second chip couples to the data-in line of a third. The data-in and data-out lines of multiple chips are therefore coupled in a daisy-chain configuration.

[0006] The IEEE 1152 bus is a newer, enhanced, version of the 1149.1 JTAG bus. References herein to a JTAG bus are intended to include both the 1149.1 and 1152 variations.

[0007] Programmable Logic Devices, herein referenced as PLDs, are commonly used as components of computer systems. These devices include a Programmable Array Logic devices (PALs), Programmable Logic Arrays (PLAs), Complex Programmable Logic Devices (CPLDs), and Field Programmable Gate Arrays (FPGAs). PLDs are typically general-purpose devices that take on a system-specific function when a function-determining, or configuration, code is incorporated within them. PLDs may store the function-determining code in fusible links, antifuses, EPROM cells, EEPROM cells including FLASH cells, or static RAM cells.

[0008] Those PLD devices which utilize static RAM cells to hold their function-determining code may be designed to automatically retrieve that code from an EEPROM on the same or different integrated circuit at system power-up. Many common FPGA devices available from Xilinx, Altera, Lucent, and Atmel are known as SRAM-based FPGAs because they store their codes in static RAM cells.

[0009] FPGAs of this type are known that can retrieve configuration code from an external EEPROM in either serial or parallel mode at system power-up. These devices are typically configured to automatically retrieve their configuration code on system power-up.

FPGAs that retrieve configuration code in serial mode can be designed to use a custom serial bus designed for loading code into an FPGA, and can be designed to use a standard serial bus such as the IIC and SPI busses although many such devices use custom serial busses. The term serial bus as used herein therefore is inclusive of IIC, SPI, and custom serial busses.

[0010] FPGAs are also known that are capable of performing a checksum verification on their configuration code when they receive it from an EEPROM. These FPGAs generate an error signal when the checksum verification fails, indicating that their configuration code might not be correct.

[0011] It is known that some EEPROM devices, including but not limited to Xilinx XC18V00 series devices, can interconnect to the JTAG bus and may be erased and programmed with a configuration code over the JTAG bus. Further, it is known that these devices can be connected to an FPGA to provide configuration code to the FPGA. It is also known that some FPGA devices can also interconnect to a JTAG bus for test or configuration purposes.

[0012] It is known that a portable programming device may connect to a JTAG bus of a board through an in-system configuration header on the board. The JTAG bus couples to at least one JTAG-configurable EEPROM on the board, that are in turn coupled to configure FPGAs on the board. A configuration system is coupled to the JTAG bus through the header; and the system is placed in a configuration mode. Configuration code is then written from the configuration system, through the header, and over the JTAG bus, into the EEPROM. Once the code is in the EEPROM, system power may be cycled; at which time the configuration code is transferred into the associated FPGA. This process is outlined in XILINX datasheet DS026 and other documents available from XILINX.

[0013] The configuration system is typically a notebook computer having configuration code for the FPGAs of the board. The configuration system also has suitable software and hardware for driving the JTAG bus of the board, together with knowledge of the JTAG bus configuration of the board.

[0014] While loading FPGA configuration code into EEPROMs of a board works well for small systems, it can pose difficulties with large systems. Large systems may have multiple boards, not all of which are connected to the same JTAG bus. Separate chains are often used because:

1. a configuration system must have knowledge of all devices in the chain in order to properly address any device on the chain; if a single chain is used the configuration system must have detailed knowledge of every board in the system.
2. large systems may, and often do, have slots permitting later addition or upgrade of peripheral devices, memory subsystems, processors, and other subsystems; additional circuitry would be required to avoid breaking a single chain at any empty slot.
3. large systems are often customized before shipment with a specific set of peripheral devices, memory subsystems, processors, and other devices; a single chain could require customized JTAG interface software for each system configuration.
4. access is faster to devices in short chains than to devices in long chains. A single board may, but need not, therefore embody more than one chain within the board.

[0015] The prior configuration process also poses difficulties when separate JTAG busses are used to load FPGA configuration code into EEPROMs of each board of a large system. For example, the multiple circuit boards of large systems are often not readily accessible for coupling of a configuration system to a configuration header without removing them from the system. Certain boards may be accessible, but only if one or more additional boards are first removed from the system. Physical access to a system by a technician also may require travel expense. In either case, substantial labor and system downtime may be required to update the FPGA configuration codes of all boards of a large system.

[0016] It is known that computer systems may have more than one data communications bus for different purposes. For example, commonly available computers have a PCI bus for communications with peripheral interface cards, one or more processor busses interfacing to each processor, and busses of other types. Complex systems may also utilize serial busses for particular purposes. For example, a complex computer system may use an IIC or SPI bus as a system management bus.

[0017] A bus bridge is a device for interconnecting busses of different types. For example, a typical personal computer utilizes at least one bus bridge between parallel busses, coupling a processor bus to a PCI bus.

[0018] A system management bus may provide an interface to system functions including, but not limited to, power supply voltage monitors, temperature sensors, fan controls, and fan speed monitors to a dedicated system management processor. The system management processor may in turn be interfaced through appropriate hardware, which may include one or more bus bridges, to other processors of the system.

[0019] In such a system, the system management processor may monitor system functions and determine if any system functions exceed limits. When limits are exceeded, the system management processor can protect the system by altering fan speeds, by instructing the system to operate in particular modes, including shutdown, or by other means known in the art.

[0020] Complex computer systems may embody multiple FPGAs and other PLDs. FPGAs may be used for customized I/O functions interfacing CPUs of the system to other devices, for communications between CPUs, and to interface devices such as fans and temperature sensors to a system management bus.

SUMMARY OF THE INVENTION

[0021] The present invention is a system having multiple, interconnected, circuit boards, several of which have at least one EEPROM to provide configuration code to an FPGA. The EEPROM devices of each such board are coupled into a JTAG bus, with a separate chain for each such board. The JTAG busses from each such board are connected to a central system-configuration point. The system-configuration point is located on a particular board that is readily accessible to a technician for connection of a configuration header.

[0022] The system-configuration point is equipped with a rotary switch for determining which of several board-specific JTAG busses are to receive configuration code information through the configuration header.

[0023] When it is necessary to update the FPGA code of boards of the system, a technician accesses the system and couples a configuration system to the configuration header. The technician then sets the rotary switch to a setting appropriate for the first board to receive configuration code. Once the switch is set, configuration code is transferred from the configuration system into the EEPROMs of the board. Once the first board has received configuration code, the switch may be reset to a setting appropriate for the next board to receive configuration code, and configuration code is transferred into that board.

BRIEF DESCRIPTION OF THE DRAWINGS

[0024] Figure 1 is a block diagram of a prior art computer system having multiple JTAG busses on multiple boards, each board having a separate configuration header;

[0025] Figure 2 is a block diagram of a computer system having multiple JTAG busses from multiple boards brought to a common system-configuration point;

[0026] Figure 3 is a block diagram of a common system-configuration point of the system of Figure 2;

[0027] Figure 4 is a flowchart of a method of configuring FPGAs of a system through a common system-configuration point.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0028] A computer system as known in the art incorporates multiple circuit boards, such as Board A 100 (Figure 1) and Board B 102 embodying FPGAs 104, 106, 107 on the boards. There may be additional boards in the system, both with and without FPGAs, the various boards being coupled together 103 as components of the system. On Board A 100, FPGA 104 is coupled to a configuration EEPROM 108, such that FPGA 104 receives its configuration code from EEPROM 108 when Board A 100 is powered-up. Similarly, FPGA 106 is coupled to a second configuration EEPROM 110. Configuration EEPROMs 108 and 110 are chained together in a JTAG bus 111, that is brought out to a configuration header 112.

[0029] When it is desired to update configuration code of one or more of the FPGAs 104 or 106 on Board A 100, a configuration system 114 is coupled through a configuration cable 116 to configuration header 112. Configuration code may then be transferred from a memory system 118 of configuration system 114, through configuration cable 116 and configuration header 112, and over the JTAG bus 111, into an EEPROM such as EEPROM 108. Once this is accomplished, power may be cycled to cause FPGA 104 to load the updated configuration code from EEPROM 108.

[0030] When it is desired to update configuration code of FPGAs on a different board, such as Board B 102, the configuration cable 116 is disconnected from configuration header 112 and coupled to an appropriate configuration header 120 of Board B, along an alternate configuration cable routing 122. The process is then repeated to update appropriate EEPROMs of EEPROMs 124 over a Board B JTAG bus 126.

[0031] The prior-art in-system FPGA configuration code update apparatus illustrated in Figure 1 requires physical access to each board of the system that is to be updated, so that the configuration cable 116 can be connected to appropriate configuration headers.

In a computer system 198 of the present invention, there are multiple circuit boards, such as Board C 200 (Figure 2) and Board D 202 embodying FPGAs 204, 206, 207. There may be

additional boards in the system, both with and without FPGAs, the various boards being coupled together 203 as components of the system. On Board C 200, FPGA 204 is coupled to a configuration EEPROM 208, such that FPGA 204 receives its configuration code from EEPROM 208 when Board C 200 is powered-up. Similarly, FPGA 206 is coupled to a second configuration EEPROM 210. Configuration EEPROMs 208 and 210 are chained together in a JTAG chain, or JTAG bus, 211.

[0032] JTAG bus 211 is brought to a common configuration point 214 which may be located on a third board, Board E 216, of the system.

[0033] Similarly, FPGAs 207 of Board D are coupled to receive configuration code from EEPROMs 218, which are coupled into a JTAG bus 220 that is also coupled to the common configuration point 214. Common configuration point 214 incorporates selection apparatus 222 and a configuration header 224.

[0034] When it is desired to update configuration code of one or more of the FPGAs 204 or 206 on Board C 200, a configuration system 230 is coupled through a configuration cable 232 to configuration header 224. The selection apparatus 222 is then set such that the Board C JTAG bus 211 is selected for programming. Configuration code may then be transferred from a memory subsystem 238 of configuration system 230 through configuration cable 232, configuration header 224, over the JTAG bus 211, and into an EEPROM such as EEPROM 208. Once this is accomplished, power may be cycled to cause FPGA 204 to load the updated configuration code from EEPROM 208.

[0035] If it is also desired to update configuration code of FPGAs on a different board, such as Board D 202, there is no need to move configuration cable 232 – it is left coupled to configuration header 224. Selection apparatus 222 is altered to designate the Board D JTAG bus 220 for programming. Configuration code is then transferred to update appropriate EEPROMs, such as EEPROMs 218, over Board D JTAG bus 220 from memory subsystem 238 of the configuration system 230.

[0036] The common configuration point 214 of a particular embodiment has a clock line buffer 300 (Figure 3) to buffer JTAG clocks 301 received through configuration header 224 and provide them to a first 302 of the multiple board-specific JTAG configuration busses. Similarly, a data line buffer 304 buffers JTAG serial data 306 and provides data to JTAG bus 302. Test Mode line 308 from configuration header 224 is routed to an enable input of a decoder device 310.

00013001 00013002 00013003 00013004 00013005 00013006 00013007 00013008 00013009 00013010 00013011 00013012 00013013 00013014 00013015 00013016 00013017 00013018 00013019 00013020 00013021 00013022 00013023 00013024 00013025 00013026 00013027 00013028 00013029 00013030 00013031 00013032 00013033 00013034 00013035 00013036 00013037 00013038 00013039 00013040 00013041 00013042 00013043 00013044 00013045 00013046 00013047 00013048 00013049 00013050 00013051 00013052 00013053 00013054 00013055 00013056 00013057 00013058 00013059 00013060 00013061 00013062 00013063 00013064 00013065 00013066 00013067 00013068 00013069 00013070 00013071 00013072 00013073 00013074 00013075 00013076 00013077 00013078 00013079 00013080 00013081 00013082 00013083 00013084 00013085 00013086 00013087 00013088 00013089 00013090 00013091 00013092 00013093 00013094 00013095 00013096 00013097 00013098 00013099 00013100 00013101 00013102 00013103 00013104 00013105 00013106 00013107 00013108 00013109 00013110 00013111 00013112 00013113 00013114 00013115 00013116 00013117 00013118 00013119 00013120 00013121 00013122 00013123 00013124 00013125 00013126 00013127 00013128 00013129 00013130 00013131 00013132 00013133 00013134 00013135 00013136 00013137 00013138 00013139 00013140 00013141 00013142 00013143 00013144 00013145 00013146 00013147 00013148 00013149 00013150 00013151 00013152 00013153 00013154 00013155 00013156 00013157 00013158 00013159 00013160 00013161 00013162 00013163 00013164 00013165 00013166 00013167 00013168 00013169 00013170 00013171 00013172 00013173 00013174 00013175 00013176 00013177 00013178 00013179 00013180 00013181 00013182 00013183 00013184 00013185 00013186 00013187 00013188 00013189 00013190 00013191 00013192 00013193 00013194 00013195 00013196 00013197 00013198 00013199 00013200 00013201 00013202 00013203 00013204 00013205 00013206 00013207 00013208 00013209 00013210 00013211 00013212 00013213 00013214 00013215 00013216 00013217 00013218 00013219 00013220 00013221 00013222 00013223 00013224 00013225 00013226 00013227 00013228 00013229 00013230 00013231 00013232 00013233 00013234 00013235 00013236 00013237 00013238 00013239 00013240 00013241 00013242 00013243 00013244 00013245 00013246 00013247 00013248 00013249 00013250 00013251 00013252 00013253 00013254 00013255 00013256 00013257 00013258 00013259 00013260 00013261 00013262 00013263 00013264 00013265 00013266 00013267 00013268 00013269 00013270 00013271 00013272 00013273 00013274 00013275 00013276 00013277 00013278 00013279 00013280 00013281 00013282 00013283 00013284 00013285 00013286 00013287 00013288 00013289 00013290 00013291 00013292 00013293 00013294 00013295 00013296 00013297 00013298 00013299 00013300 00013301 00013302 00013303 00013304 00013305 00013306 00013307 00013308 00013309 00013310 00013311 00013312 00013313 00013314 00013315 00013316 00013317 00013318 00013319 00013320 00013321 00013322 00013323 00013324 00013325 00013326 00013327 00013328 00013329 00013330 00013331 00013332 00013333 00013334 00013335 00013336 00013337 00013338 00013339 00013340 00013341 00013342 00013343 00013344 00013345 00013346 00013347 00013348 00013349 00013350 00013351 00013352 00013353 00013354 00013355 00013356 00013357 00013358 00013359 00013360 00013361 00013362 00013363 00013364 00013365 00013366 00013367 00013368 00013369 00013370 00013371 00013372 00013373 00013374 00013375 00013376 00013377 00013378 00013379 00013380 00013381 00013382 00013383 00013384 00013385 00013386 00013387 00013388 00013389 00013390 00013391 00013392 00013393 00013394 00013395 00013396 00013397 00013398 00013399 00013400 00013401 00013402 00013403 00013404 00013405 00013406 00013407 00013408 00013409 00013410 00013411 00013412 00013413 00013414 00013415 00013416 00013417 00013418 00013419 00013420 00013421 00013422 00013423 00013424 00013425 00013426 00013427 00013428 00013429 00013430 00013431 00013432 00013433 00013434 00013435 00013436 00013437 00013438 00013439 00013440 00013441 00013442 00013443 00013444 00013445 00013446 00013447 00013448 00013449 00013450 00013451 00013452 00013453 00013454 00013455 00013456 00013457 00013458 00013459 00013460 00013461 00013462 00013463 00013464 00013465 00013466 00013467 00013468 00013469 00013470 00013471 00013472 00013473 00013474 00013475 00013476 00013477 00013478 00013479 00013480 00013481 00013482 00013483 00013484 00013485 00013486 00013487 00013488 00013489 00013490 00013491 00013492 00013493 00013494 00013495 00013496 00013497 00013498 00013499 00013500 00013501 00013502 00013503 00013504 00013505 00013506 00013507 00013508 00013509 00013510 00013511 00013512 00013513 00013514 00013515 00013516 00013517 00013518 00013519 00013520 00013521 00013522 00013523 00013524 00013525 00013526 00013527 00013528 00013529 00013530 00013531 00013532 00013533 00013534 00013535 00013536 00013537 00013538 00013539 00013540 00013541 00013542 00013543 00013544 00013545 00013546 00013547 00013548 00013549 00013550 00013551 00013552 00013553 00013554 00013555 00013556 00013557 00013558 00013559 00013560 00013561 00013562 00013563 00013564 00013565 00013566 00013567 00013568 00013569 00013570 00013571 00013572 00013573 00013574 00013575 00013576 00013577 00013578 00013579 00013580 00013581 00013582 00013583 00013584 00013585 00013586 00013587 00013588 00013589 00013590 00013591 00013592 00013593 00013594 00013595 00013596 00013597 00013598 00013599 00013600 00013601 00013602 00013603 00013604 00013605 00013606 00013607 00013608 00013609 00013610 00013611 00013612 00013613 00013614 00013615 00013616 00013617 00013618 00013619 00013620 00013621 00013622 00013623 00013624 00013625 00013626 00013627 00013628 00013629 00013630 00013631 00013632 00013633 00013634 00013635 00013636 00013637 00013638 00013639 00013640 00013641 00013642 00013643 00013644 00013645 00013646 00013647 00013648 00013649 00013650 00013651 00013652 00013653 00013654 00013655 00013656 00013657 00013658 00013659 00013660 00013661 00013662 00013663 00013664 00013665 00013666 00013667 00013668 00013669 00013670 00013671 00013672 00013673 00013674 00013675 00013676 00013677 00013678 00013679 00013680 00013681 00013682 00013683 00013684 00013685 00013686 00013687 00013688 00013689 00013690 00013691 00013692 00013693 00013694 00013695 00013696 00013697 00013698 00013699 00013700 00013701 00013702 00013703 00013704 00013705 00013706 00013707 00013708 00013709 00013710 00013711 00013712 00013713 00013714 00013715 00013716 00013717 00013718 00013719 00013720 00013721 00013722 00013723 00013724 00013725 00013726 00013727 00013728 00013729 00013730 00013731 00013732 00013733 00013734 00013735 00013736 00013737 00013738 00013739 00013740 00013741 00013742 00013743 00013744 00013745 00013746 00013747 00013748 00013749 00013750 00013751 00013752 00013753 00013754 00013755 00013756 00013757 00013758 00013759 00013760 00013761 00013762 00013763 00013764 00013765 00013766 00013767 00013768 00013769 00013770 00013771 00013772 00013773 00013774 00013775 00013776 00013777 00013778 00013779 00013780 00013781 00013782 00013783 00013784 00013785 00013786 00013787 00013788 00013789 00013790 00013791 00013792 00013793 00013794 00013795 00013796 00013797 00013798 00013799 00013800 00013801 00013802 00013803 00013804 00013805 00013806 00013807 00013808 00013809 00013810 00013811 00013812 00013813 00013814 00013815 00013816 00013817 00013818 00013819 00013820 00013821 00013822 00013823 00013824 00013825 00013826 00013827 00013828 00013829 00013830 00013831 00013832 00013833 00013834 00013835 00013836 00013837 00013838 00013839 00013840 00013841 00013842 00013843 00013844 00013845 00013846 00013847 00013848 00013849 00013850 00013851 00013852 00013853 00013854 00013855 00013856 00013857 00013858 00013859 00013860 00013861 00013862 00013863 00013864 00013865 00013866 00013867 00013868 00013869 00013870 00013871 00013872 00013873 00013874 00013875 00013876 00013877 00013878 00013879 00013880 00013881 00013882 00013883 00013884 00013885 00013886 00013887 00013888 00013889 00013890 00013891 00013892 00013893 00013894 00013895 00013896 00013897 00013898 00013899 00013900 00013901 00013902 00013903 00013904 00013905 00013906 00013907 00013908 00013909 00013910 00013911 00013912 00013913 00013914 00013915 00013916 00013917 00013918 00013919 00013920 00013921 00013922 00013923 00013924 00013925 00013926 00013927 00013928 00013929 00013930 00013931 00013932 00013933 00013934 00013935 00013936 00013937 00013938 00013939 00013940 00013941 00013942 00013943 00013944 00013945 00013946 00013947 00013948 00013949 00013950 00013951 00013952 00013953 00013954 00013955 00013956 00013957 00013958 00013959 00013960 00013961 00013962 00013963 00013964 00013965 00013966 00013967 00013968 00013969 00013970 00013971 00013972 00013973 00013974 00013975 00013976 00013977 00013978 00013979 00013980 00013981 00013982 00013983 00013984 00013985 00013986 00013987 00013988 00013989 00013990 00013991 00013992 00013993 00013994 00013995 00013996 00013997 00013998 00013999 00014000

[0037] The decoder device 310 also receives a binary select code from binary coded selection switch 312, and passes test mode line information from header test mode line 308 to a selected board-specific test mode line. When the first 302 of the multiple board-specific JTAG configuration busses is selected, decoder device 310 passes test mode line information to the test mode line 312 of JTAG bus 302. Similarly, should the second 314 of the multiple board-specific JTAG configuration busses be selected, decoder device 310 passes test mode line information to the test mode line 316 associated with the second 314 JTAG configuration bus.

[0038] JTAG serial data out line 318 passes from first 302 of the multiple board-specific JTAG configuration busses to a read multiplexor 320. When the first 302 of the busses is selected, this read data is passed on to the JTAG data out line 322 of header 224. Similarly, should the second 314 JTAG bus be selected, its bus-specific JTAG serial data output line 324 is coupled through the read multiplexor 320 to JTAG data out line 322. Resistors 326, 328, and 330 are provided to ensure that the header JTAG clock, JTAG data input, and JTAG test mode select lines are at defined levels whenever no configuration system is connected to the header.

[0039] The invention has been described with reference to a read multiplexor for selecting a board JTAG port to be read to the common configuration point configuration header 224. It is anticipated that a decoder and tristate gates would be operable in place of the read multiplexor shown.

[0040] With reference to Figures 2, 3 and 4, when it is desired to program, or to change code for, an FPGA without removing the affected board from the system, a technician couples 400 (Figure 4) a configuration system 230 to the configuration header 224. The technician then sets 402 selection switch 312 to designate a particular JTAG bus of the system having the EEPROM associated with the FPGA. The technician then starts 404 a configuration program on the configuration system, and designates an FPGA code file appropriate for the affected JTAG bus.

[0041] The configuration system addresses the selected JTAG bus and determines 406 the JTAG bus configuration, including the number and types of devices on the bus. This is accomplished in part through using the JTAG "GET_DEVICE_ID" command, that returns a code indicative of the type of each device connected to the JTAG bus. This is compared 408 against information in the FPGA code file to ensure that switch 312 is correctly set and that code is not programmed into a board not compatible with it. Should the code be

incompatible with the selected board, an error is declared 410. These steps verify compatibility of the code file with the selected circuit board.

[0042] In an alternative embodiment, instead of, or in addition to, comparing JTAG bus configuration against information in the code file, board identification information is read from an EEPROM located on the board. This board identification information is used to verify compatibility of the code file with the board, and may also be used to select appropriate FPGA code from among several FPGA codes contained within the code file.

[0043] Next, the configuration system erases 412 one or more EEPROMs of the board that are attached to the JTAG bus. More than one EEPROM may be erased should the FPGA code file contain code for more than one FPGA of the board. Then, the configuration system writes 414 new code into the erased EEPROM(s). Finally, the configuration system checks 416 for errors in the EEPROM writing process and declares an error 418 if any error occurred and was reported by an EEPROM. If the code file was written correctly into the EEPROMs of the board, the technician is so notified. The technician may then reset 420 the selection switch to indicate the next JTAG bus to be programmed, if any, and restart the configuration program to program that JTAG bus in the same way as described for the first JTAG bus.

[0044] After all JTAG bus of the system have been programmed, the technician power-cycles 422 the system, such that each FPGA of the system reloads its code from the associated EEPROMs at power-up.

[0045] While the invention has been described with reference to selection apparatus comprising a binary coded switch and decoder as illustrated in Figure 3, it is anticipated that it would be operable with alternative circuitry.

[0046] While Figure 3 illustrates a common configuration point having three board-specific JTAG busses to avoid clutter, the invention is applicable to other numbers of board-specific JTAG busses. A particular embodiment of the invention embodies ten board-specific JTAG busses.

[0047] It is also anticipated that the invention would be operable with electronic selection apparatus. Such electronic selection apparatus could take the form of an IIC or JTAG addressable register, operating under control of the configuration system to automate selection of particular board-specific JTAG busses of the system.

[0048] While the invention has been described with reference to a separate configuration system coupled to a configuration header of a common configuration point, it is

anticipated that the invention would be operable should the system incorporate JTAG interface hardware in place of, or in addition to, the configuration header illustrated. In this way, need of a separate configuration system could be eliminated.

[0049] The invention has been described with reference to board specific JTAG busses. It is anticipated that one or more boards of a system may have more than one such JTAG bus on the board.

[0050] While the invention has been particularly shown and described with reference to a preferred embodiment thereof, it will be understood by those skilled in the art that various other changes in the form and details may be made without departing from the spirit and scope of the invention. It is to be understood that various changes may be made in adapting the invention to different embodiments without departing from the broader inventive concepts disclosed herein and comprehended by the claims that follow.